SERVICE ORIENTED ARCHITECTURE REFERENCE ARCHITECTURE BLUEPRINT

Edison[1], Virginia Tulenan[1], and Ford Lumban Gaol[2]

[1]Bina Nusantara University Graduate Program, Jakarta, Indonesia
edison17999@yahoo.sg, virginia_tulenan@yahoo.com

[2]STMIK JIBES, Jakarta, Indonesia
ford.gaol@jibes.ac.id

Abstract

Service Oriented Architecture (SOA) is a framework for integrating business processes and supporting IT infrastructure as secure, standardized components-services-that can be reused and combined to address changing business priorities. Services are the building blocks of SOA and new applications can be constructed through consuming these services and orchestrating services within a business process. In SOA, services map to the business functions that are identified during business process analysis. Upon a successful implementation of SOA, the enterprise gain benefit by reducing development time, utilizing flexible and responsive application structure, and following dynamic connectivity of application logics between business partners. This paper presents SOA reference architecture blueprint as the building blocks of SOA which is services, service components and flows that together support enterprise business processes and the business goals.

Keywords: SOA, SOA reference architecture, services.

Introduction

Service-oriented architecture (SOA) is an architectural styles that modularized information system into services. With SOA, these important programs become business services. With one single business service for a given function that get used everywhere in organization. When business policy need to be changed, it can be changed in one place and because the same service is used everywhere, the consistency will be keep throughout the organization. SOA enables businesses to make business decisions supported by technology instead of making business decisions determined by or constrained by technology. With SOA, business gets to focus more on business and less on technology; this is the reason why business managers are going to like SOA.

Service-oriented architecture is all about building standard interfaces to access different business functions that are exposed by various core business backend systems. These functions could essentially be those that are frequently invoked by other business systems within the enterprise ecosystem [1].

SOA enables business managers and IT to talk in business term that both sides understand. Without SOA, the IT developers and business manager typically use very different words to describe the process of creating programs; and that's make business peoples have to do extra learning for the IT terms that will consume time and effort.

Businesses keep changing and requests for new programs keep coming. What is new and different about business is the idea that businesses do not have to keep reinventing wheel. The idea behind a service oriented architecture is that the idea that can organize programs for easy reuse, for easy maintenance and supporting, for coherent, consistent results across organization, and for easily sharing data and resources.

Service oriented architecture is to create flexible structure that will turn key IT software asset into reusable services that can be use no matter how the business changes. The aims of this Service Oriented Architecture paper are to develop an SOA blueprint that provide reference architecture of integration to extend the utilization of the functionality delivered by application programs, the concept to provides agility toward changing business needs, the solution to speed up delivery of business application by leveraging reusable services; and architecture to create scalability, consistency and maintainability of information technology systems throughout the organization. Meanwhile, the benefits that can be achieved with Service Oriented Architecture by using

this blueprint guidance are integration of IT systems across siloed applications, increased flexibility and added agility to business-aligned applications, achieve operational excellence and assist business leader decision making, and to minimize the effort and time for development hence reduce cost.

The remainder of this paper is organized as follows: Section 2 present a SOA modeling and the methodology for SOA modeling. Section 3 discusses the SOA reference architecture and its component layers. Section 4 illustrate the case study at PT. Total Bangun Persada Tbk (TBP) and the design approach blueprint for solving TBPs problems using SOA reference architecture and Section 5 concludes the paper.

SOA Modeling

SOA is about bridging the gap between business and IT through a set of business aligned IT services using a set of design principles, patterns and techniques. Methodology discuss here is the key needed for analysis and design required to build a Service Oriented Architecture. The methodology stresses the importance of addressing the techniques required for identification, specification and realization of services, the services flow and composition, as well as the enterprise-scale components needed to realize and ensure the quality of services required of an SOA.

The methodology starts with modeling. A valid SOA conceptual model is based on an architectural style that defines an interaction model between three primary parties:

- The service provider

The service provider publishes a service description and provides the implementation for the service. Service provider is the network addressable entity that accepts and executes requests from consumers. The service provider can be component, or other type of software system that fulfills the service consumer's requirements.

- A service consumer

A service consumer can use the Uniform Resource Identifier (URI) for the service description directly, or it can find the service description in a service registry and bind then invoke the service. Service consumer looks for service to execute a required function. The consumer can be an application, other service or other type of software module that needs the service.

- Service registry

Service registry is a directory which can be accessible through network and contain available services. Its main function is to store and publish service descriptions from providers and deliver these descriptions to interested service consumers.

- The service broker

The service broker provides and maintains the service registry

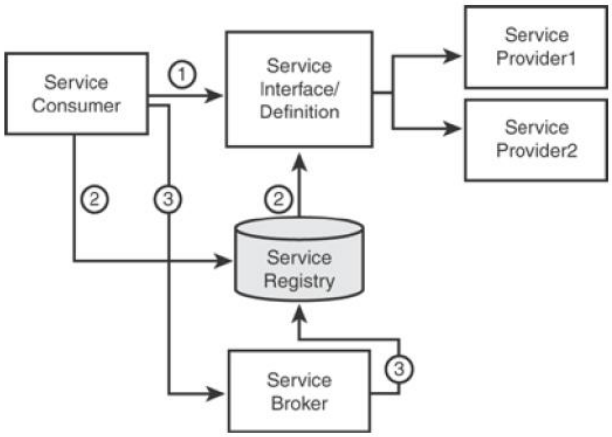


Figure 1. SOA Remote Service Strategy Pattern

Web services are one of aspect that is important as part of SOA. It enables technology independence by removing the technology dependency between the provider and requestor platforms. Web services allows links between systems to be resolved at run-time, availability and functionality of a service to be discovered at run-time rather than design time and enables systems to call functions across the Internet [3].

XML is the important base of Web service and use it to communicate between applications. In other words, XML is a universal language that unites and used to communicate and exchange information. XML describes the data in an application-independent manner, and Web services use this technology to enable the sharing of distributed process between heterogeneous computing environments [4].

SOA Reference Architecture

An SOA template consists of partially layered architecture of composite services that align with business processes. When developing an SOA, it is necessary to combine a top-down business-driven approach with a bottom-up approach leveraging legacy investments. The service-oriented modeling approach provides modeling, analysis, design techniques, and activities to define the foundations of an SOA. It helps by defining the elements in each of the SOA layers and making critical architectural decisions at each level. The lower layers (services, service components and operational layer) are

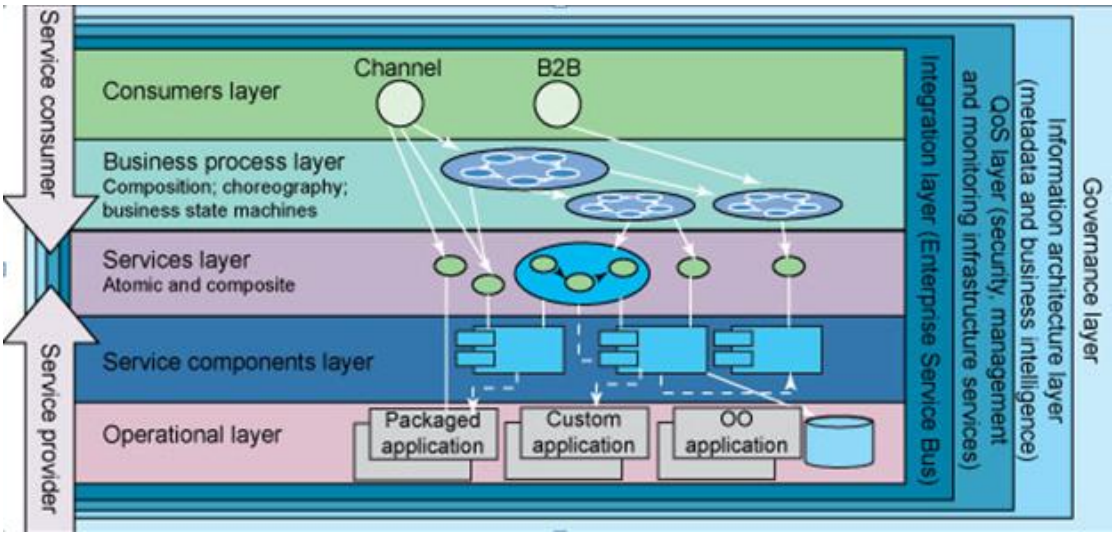


Figure 2. Layers of SOA Reference Architecture [8]

concerns for the provider, and the upper ones (services, business processes, and consumers) are concerns for the consumer. The horizontal layers relate to overall functionality of SOA solution. The vertical layers are to support various concerns that happened across the functional layers.

The SOA reference architecture covers all three SOA component areas by providing approaches, requirements and design patterns wherever possible. SOA reference architecture establish the building blocks of SOA which is services, service components and flows that together support enterprise business processes and the business goals [2].

a. Operational layer

This layer includes all custom or packaged application assets running in an IT operating environment of the applications portfolio for supporting business activities. The assessment can help to reduce budget for new initiative and development of new business critical services. This layer consist a number of existing software systems including existing package applications and solutions such as ERP and CRM packages, existing database, existing transaction processing systems and legacy applications and systems. Operational layer define all information technology assets which is used by the enterprise to perform its business process such as application portfolio and infrastructure environment.

Universal Modeling Language (UML) is use to define the scope of SOA. With UML, the architecture for the area of enterprise business process is draw as service oriented analysis and design for modeling, analyzing, designing, and producing an SOA that aligns with business analysis, processes, and goals. In this layer, all the business processes from all the applications system within organization are drawn into diagrams according to UML, then the diagrams can be use to establish the services needed for the SOA.

b. Service components layer

This layer contains programs other than programs in operational system layer to help performs the services. The programs are the extract of the applications in operational layer, which gathered from the diagrams of the UML. The programs wrap the programs in operational layer as a component to create services. A service component may realize one or more services. The service component provides an enforcement point for service realization to ensure quality of service (QoS) and compliance to service level agreements. A service component can be created based on multiple applications and may separated from operational layer which in implementation all crucial accesses and integration are not being exposed to consumer.

c. Services layer

This layer consists all of the services define in SOA. A service is considered to be an abstract specification of business aligned IT function collections. This specification provides users with sufficient detail to invoke the business functions exposed by provider of the service. Ideally, this activity can be done in platform independent manner. The service specification may also include a policy document, SOA management descriptions, and attachment that categorize or show service dependencies.

Services is business functions of service component which prepared by provider to support

consumers business processes through Enterprise Service Bus (ESB). For example, the logistic system need business function from POCA to produce important information needed for order and delivery from supplier. The appliance of this layer is web services and XML.

d. Business process layer

This layer defined compositions and choreographies of services exposed in services layer. At this layer, service composition is used to combine certain services from service layer thus it would have the necessary flow to run a business process. If a process need certain business function from several services that is taken from some applications components then service composition will design the service flow, thus create an application that support use cases and business processes.

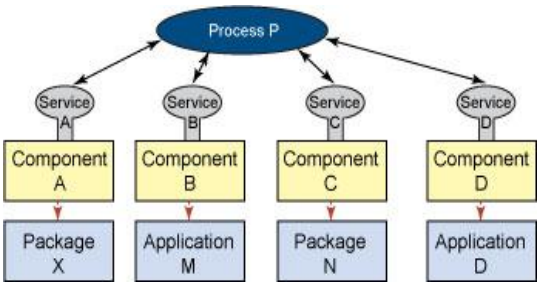


Figure 3. Business Process Layer [8]

e. Consumer layer

The business process layer communicates with the consumer layer also called the presentation layer to communicate inputs and results from the various people who use the system such as end users, decision makers, system administrators through Web portals or business-to-business (B2B) programs. This layer has an ability to supply IT functions and the data needed by consumer to fulfill the business process. This layer also prepare interface to communicate between applications. For example, search data using input and search button need service based application. An AJAX application that can send XML without have to refresh the web browser can increase consumer layer interaction.

f. Integration layer

Integration layer enables the integration of services through the introduction of a reliable set of capabilities, such as intelligent routing, protocol mediation, and other transformation mechanisms, often described as the Enterprise Service Bus (ESB). Web Services Description Language (WSDL) specifies a binding, which implies a location where the service is provided. On the other hand, an ESB provides a location independent mechanism for integration.

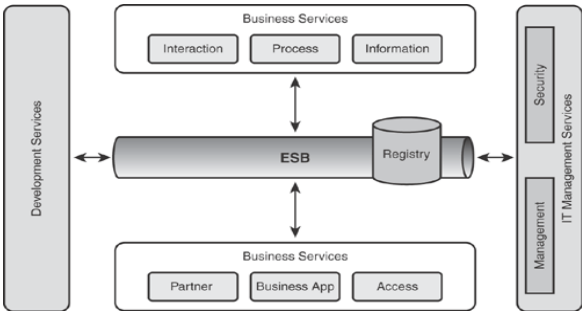


Figure 4. Logical View of ESB in Integration Layer

g. Quality of service layer

The QoS layer provides the capabilities required to monitor, manage, and maintain QoS such as security, performance, and availability. This is a background process through sense-and-respond mechanisms and tools that monitor the health of SOA applications. At this layer, all registered services would be monitored from its performance, availability and security level to create KPI which can monitor SOA performance. The resulted KPI can create good QoS standard for SOA development life cycle.

h. Information architecture and BI layer

This layer contains functions that concerned with the transformation and management of data. At this layer, architectural design of represented data through SOA can be used to create Business Intelligent (BI) through data marts and data warehouses. This layer is applicable for industries that supply crucial services.

i. Governance layer

SOA governance is strategic planning for SOA lifecycle. This layer manages all operational procedures and information related to SOA itself. Governance layer supply instruction and procedure for decision making related to SOA and arrange all aspects derived from SOA solution such as capacity, performance, and security and monitoring. By using SOA governance, application development will be more effective by service reuse. The design strategy for an SOA does not start from the bottom-up as is often the case with a Web services-based approach. Web services are a tactical implementation of SOA. A number of important activities and decisions exist that influence not just integration architecture but enterprise and application architectures as well.

The SOA Approach

As an example of a case study, this paper consider Indonesia's premier building contractor, PT. Total Bangun Persada Tbk (TBP) and have built projects ranging from small factories to high rise building and mega projects all over Indonesia since 1969.

This company has several systems available for business process support located in every project sites and systems that running at the head office. The system that control main business process is called Project Operation and Control Application (POCA).

POCA is a web based ERP system which implemented in every project sites and in every head office for financial works, logistic supply, cash operations and evaluation reports. The current version of POCA is 2.0 built on Microsoft .NET 3.5 platform serve by Internet Information Service (IIS) and use by Microsoft SQL Server 2008 for database management system.

At project sites, POCA is used for maintaining all the transactions of estimations of project's tender, payments, logistic and purchasing, supply control, and events related to the project. Every project sites has different functioning modules which depend on requirement of the project. With different modules customization, POCA is responsible to manage tender work flow from project start until project's completion at project sites.

Figure 5 shows the architecture of POCA's system that running at the company. At project sites, POCA is used for maintaining all the transactions of estimations of project's tender, payments, logistic and purchasing, supply control, and events related to the project. At the head office, POCA is used for the requirement of all project evaluation happened in every project sites. POCA at the head office has its own web server and database with the same specifications as project sites. The online traffic information is done from project site to head office by sending update or new information with upload function mechanism.

The problem is the synchronized database between head office and project sites that create larger size of database in head office rather than database in every project site and not to mention data redundancy. The process of data transformation from the main office to the project site must go through repetitive and unnecessary processes and all the important systems such as logistic system, accounting system, cash operation system have different platform and required Data Transformation System (DTS) to store data in its' own databases. The work of transforming data needs some times to be done and same mechanisms owned by other systems thus make it difficult to produce real time reports.

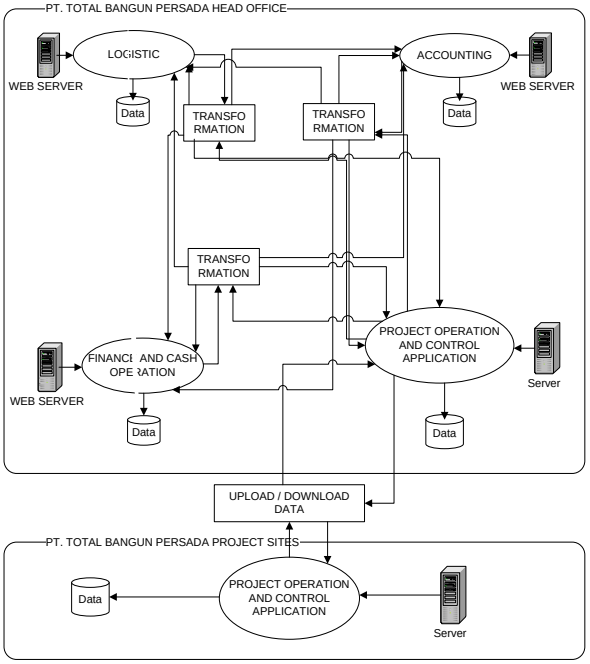


Figure 5. POCA Case Study Architecture

A lot of boundaries in systems make it difficult for IT development and business improvement. All of the problems need solution that can integrate the systems and provide services that can decrease the cost of application programs development and reduce redundancy of the systems for easy and cheaper maintenance. SOA is the architecture that can provide the architecture to solve the problems and boundaries and to improve work efficiency for the next level of business process opportunities.

Figure 6 shows the design approach using SOA reference architecture to solve TBP current applications issues and problems. The operational layer in this case study include all custom applications, packaged applications, legacy systems, transaction-processing systems, and the various databases such as logistic system, project operation and control application system, finance and cash operation system, accounting system and other systems. The components from each system then defined in the service component layer. At this layer, an implementation frontage that aggregates functionality from multiple, possible disparate, operational systems while hiding the integration and access complexities from the service that is exposed to the consumer is handled by the service component layers. Each service in the service component layer then extracted into services which is defined in the service layer. The services are defined in such a way

that they are accessible to and can be invoked by channels and consumers independent of implementation and the transport protocol.

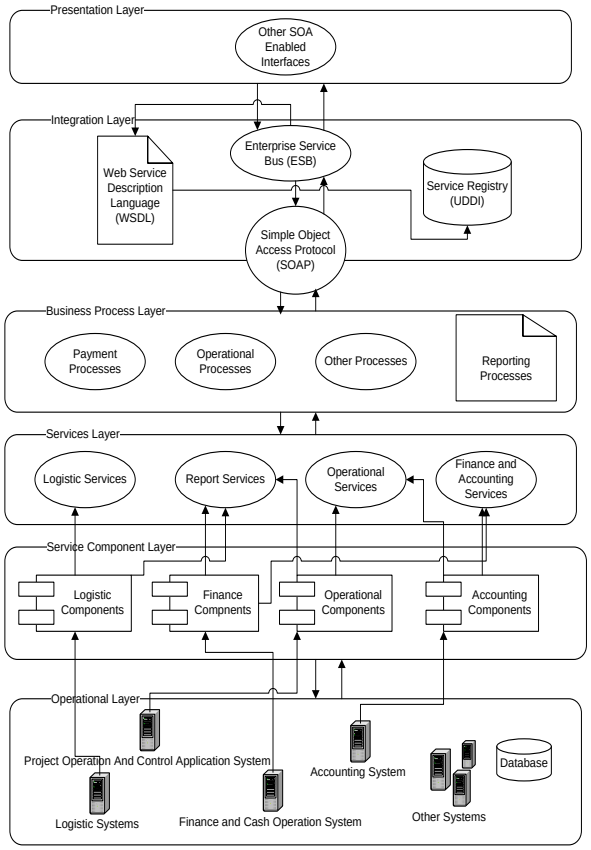


Figure 6. The SOA Reference Architecture Approach

Next, the business process layer then represents the services into processes as an orchestration or a composition of loosely coupled services—leveraging the services represented in the services layer. The layer is also responsible for the entire lifecycle management of the processes along with their orchestration, and choreography based on the business process at this case TBPs business process. Processes represented in this layer are the connection medium between business requirements and their manifestation as IT-level solutions. The data and information flow between steps within each process is also represented in this layer.

The next layer is integration layer. At this layer the capability for service consumers to locate service providers and initiate service invocations is provided. The key nonfunctional requirements such as security, latency, and quality of service between adjacent layers in the reference architecture are implemented by the architecture building blocks in this layer and defined by Enterprise Service Bus (ESB). The ESB connect various types of middleware, repositories of metadata definitions, registries and interfaces of every kind. Simple Object Access Protocol (SOAP) is a protocol that uses the XML to describe the data and Hyper Text Transfer Protocol (HTTP) to transmit application data. The Web Service and the client application must agree upon a common protocol to facilitate communication.

SOAP is a standard communication protocol for interchanging information in a structured format in a distributed environment. Messaging is an example of information exchange between a client and a web service. The calls made by a client application to a web method and the data returned by a web method to the client are the messages that are actually exchanged. A SOAP packet is created when a web client makes a call to the web method. This message contains the name of the web method and the parameters that are needed for making a call to the web method in XML format. The Web method is invoked based on the information available in the SOAP Packet. [6]

Universal Description, Discovery and Integration (UDDI) are a platform independent framework functioning like a directory that provides mechanism to locate and register web services on the internet. The web service provider makes the web service available to the consumer by describing the web service using a WDSL document and then registering the Web service in the UDDI Directory. The UDDI Directory contains pointers to the Web service and the WDSL document for the Web service. After this is done the Client Applications can discover the Web service using the UDDI Directory.

The UDDI specification calls for three elements as given below:

- White Pages
  These provide business contact information
- Yellow Pages
  These organize Web services into categories like usage billing service, authorization service and so on.
- Green Pages
  These Pages provide detailed technical information about individual services.

Web Service Discovery Language (WSDL) is a markup language that describes the web service. In order to use this Web service, the Client application developers need to know the methods exposed by the Web service and the parameters to be passed to these methods. It is imperative that access to these methodologies is available at development time and it is just this need that WSDL addresses. [6]

The SOA solution needed by TBP is designed until layer of integration because TBP is a

manufacturing company, all of the other layers such as QoS, data integration, and governance layer are mean to increase the SOA performance for company that provide services as business for customer and use SOA as the main tool.

Conclusion

This paper addressed SOA reference architecture in designing SOA blueprint for Indonesia's construction company PT. Total Bangun Persada Tbk (TBP). The Service oriented architecture is used to create flexible structure that will turn key IT software asset into reusable services that can be use no matter how the business changes. SOA integrates IT systems across siloed applications and increased flexibility and added agility to business-aligned applications. SOA provides a new blueprint to solve software reuse and enterprise information system integration which publish business functionality in the form of programming and accessible software services, other application program can use these services by published and discoverable interfaces. In this blueprint, the layers planned on TBPs until integration layer as the starting architecture for implementing SOA. This planned because TBP is not an organization that provide services to customer, instead product of manufacturing. The SOA solution will be used as back-office services to increase internal business processes. As SOA is getting mature, in the future the QoS, data integration, and governance will be needed to improve the SOA solutions.

REFERENCES

[1] W. Lam, V. Shankararaman, (2007), *Enterprise Architecture and Integration: Methods, Implementation, and Technologies*, Information Science Reference, Hershey, New York

[2] Bieberstein, N., Laird, R.G., Jones, K., Mitra, T. (2008). *Executing SOA: A practical guide for the Service-Oriented Architect*. (1st edition), IBM Press

[3] W3C Web Services Architecture Group (2004). "Web Services Architecture Requirements". Editors: Austin, D., Barbir, A., Ferris, C. and Garg, S., Available at: http://www.w3.org/TR/2004/NOTE-wsa-reqs-20040211/

[4] Lucky. (2008). XML Web Service: *Aplikasi Desktop, Internet & Handphone*, Jasakom, Jakarta

[5] Brown, Paul C. (2008), *Implementing SOA: Total Architecture in Practice*. (1st edition), Addison-Wesley Professional

[6] Exforcis. Inc (2005). "Web Services - SOAP, WDSL, Disco and UDDI", Available at: http://www.exforsys.com/tutorials/vb.net-2005/web-services-soap-wsdl-disco-and-uddi.html

[7] Zhang, T., Ying, S., Cao, S., Zhang, J., Wei, J., "A modeling approach for Service-Oriented Architecture, in International Federation for Information Processing", (2006), Volume 205, Research and Practical Issues of Enterprise Information Systems, Boston, pp.63-72.

[8] IBM (2007). "Design an SOA solution using a reference architecture", Available at: http://www.ibm.com/developerworks/library/ar-archtemp/